
Rule DAS200: VOLUME PROVIDING WORST OVERALL PERFORMANCE TO CRITICAL WORKLOAD

Finding: The identified volume provided the worst overall performance to the "loved one" workload during the entire measurement period. RULE DAS200 is quite similar to RULE DAS100; RULE DAS200 applies only to DASD devices accessed by critical (or "loved one") workload, while RULE DAS100 applies to **all** DASD devices.

Impact: This finding will have a HIGH IMPACT on the performance of the "loved one" workload.

Logic flow: This is a basic rule finding; there are no predecessor rules.

Discussion: CPExpert determines the average device response time, by device type, for each measurement interval. A "device type" for this purpose is any unique device type (e.g., IBM-3380 or IBM-3390), with the device type modified to reflect whether the device is cached, is a Parallel Access Volume (PAV), or is a paging device.

The purpose of determining the average device response time, by device type, is the underlying principle that there is little point in analyzing a particular device if its response time is better than average. Rather, the most improvement potential resides with devices whose response time is worse than average.

CPExpert selects a device in each measurement interval for further analysis if the device response time exceeds the average for its device type **and** the device was referenced by the "loved one" workload.

CPExpert consolidates information in various SMF records to build a model of the I/O configuration. This model includes utilization and queuing information for all channel paths, controllers, and devices. In creating the model, CPExpert:

- C Processes RMF Type 70 records to identify the systems that are in the sysplex.
- C Processes RMF Type 73 records to identify the physical channels that are associated with each system, and the type of channel (e.g., ESCON, FICON-Bridge, FICON-Native, etc.). Additionally, the physical and LPAR channel busy time is acquired for each system.

-
- C Processes RMF Type 78 records to obtain the logical control units associated with each system, and the channels associated with each logical control unit. Additionally, controller busy and director port busy times are acquired.
 - C Processes RMF Type 74 records to obtain devices associated with each logical control unit. Device performance characteristics are also acquired from the Type 74 records.

The result from the above is a record for each device, containing information about the devices; and the logical control units, channels, and systems that are associated with each device.

CPEXpert constructs a frequency distribution of all devices whose response is worse than average for the type of device, weighted by the number of I/O operations executed by the device. This yields a weighted measure of the potential performance improvement that might be achieved for each device. This frequency distribution is sorted descending, to yield an ordered list of the devices with the most improvement potential. This ordered list represents an "ordered intensity of access" distribution of the devices.

CPEXpert selects the top devices from the ordered list of devices referenced by the "loved one" workload. These represent the devices with the most improvement potential with respect to the "loved one" workload. CPEXpert reports information about the top devices from the list, by sysplex and by system (see Rule DAS000 and Rule DAS050 for additional information about this information).

Detailed information regarding the "worst" device is extracted and reported for each measurement interval. (RULE DAS290 reports summary results for the remaining top devices.)

Please refer to Section 5 for a discussion regarding some of the limitations of the technique of associating workload to device performance. In brief summary, the Type 30 information is not synchronized with the Type 70(series) information. Consequently, the analysis is approximate, rather than precise.

Exhibit DAS200-1 provides a sample output resulting from the analysis. The VOLSER and device number of the "worst" performing device are identified in the narrative. Information is provided about the overall average I/O rate and the device utilization for the entire measurement period being analyzed.

RULE DAS200: VOLUME WITH WORST OVERALL PERFORMANCE

VOLSER RSA002 (device 0194) provided CICS with the worst overall performance during the entire measurement period (13:59, 23JUL1991 to 16:59, 23JUL1991). This pack had an overall average of 17.2 I/O operations per second, was busy processing I/O for an average of 47% of the time, and had I/O operations queued for an average of 20% of the time. The following summarizes significant performance characteristics of VOLSER RSA002:

MEASUREMENT INTERVAL	I/O RATE	I/O RESP	--AVERAGE PER SECOND DELAYS--	MAJOR PROBLEM
			CONN DISC PEND IOSQ	
15:14-15:29,23JUL1991	16.9	0.665	0.096 0.288 0.073 0.208	QUEUEING
15:29-15:44,23JUL1991	19.0	0.825	0.104 0.347 0.101 0.273	QUEUEING
15:44-15:59,23JUL1991	18.5	0.737	0.103 0.350 0.053 0.231	QUEUEING
15:59-16:14,23JUL1991	16.8	0.688	0.096 0.298 0.061 0.233	QUEUEING
16:14-16:29,23JUL1991	19.9	0.793	0.108 0.354 0.058 0.273	QUEUEING
16:29-16:44,23JUL1991	19.6	0.874	0.102 0.356 0.105 0.311	QUEUEING
16:44-16:59,23JUL1991	17.5	0.669	0.098 0.320 0.061 0.190	QUEUEING

VOLUME WITH WORST OVERALL PERFORMANCE

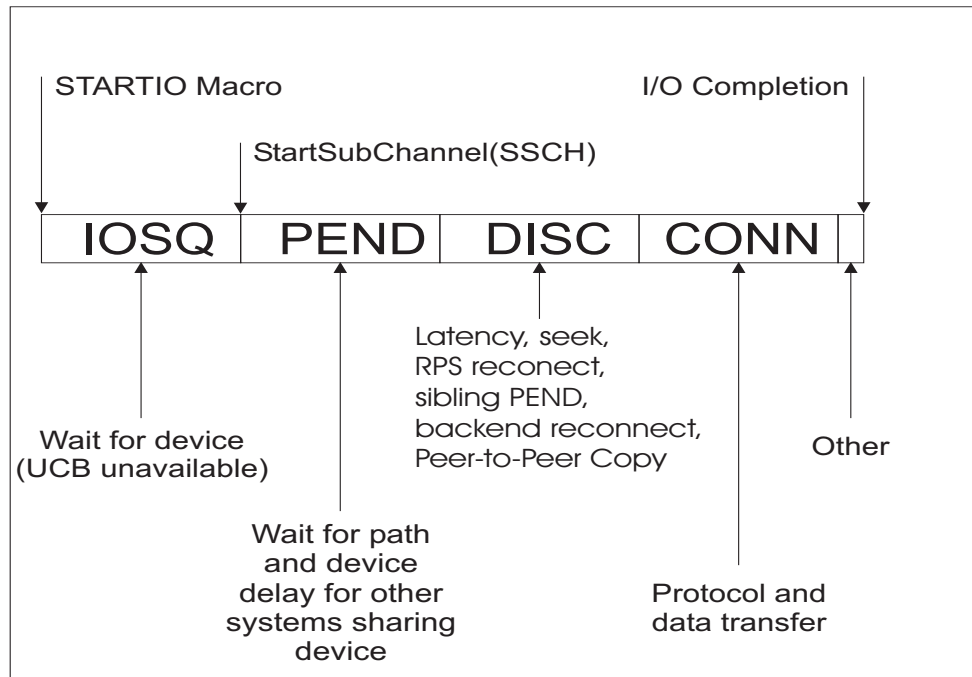
EXHIBIT DAS200-1

As shown in Exhibit DAS200-1, CPExpert provides a summary for each measurement interval, showing the average I/O rate for the interval, and the **average delay time per second** during the interval (the total is shown as **I/O RESP** in Exhibit DAS200-1). The average delay time per second effectively reflects the percent of each second (shown in milliseconds) in which the associated delay occurred.

CPExpert determines the major cause of device response delays (simply dividing each potential delay by the total device response time). The result from these calculations is evaluated to determine whether any area significantly predominates the I/O response time. If so, the respective area is listed as the major problem with the volume during the interval being analyzed.

From a high-level view, there are four key measures of DASD performance: IOS Queue (IOSQ) time, PENDING (PEND) time, disconnect (DISC) time, and connect (CONN) time. CPExpert provides information on these four key measures, and identifies the major cause of response delay.

The following figure illustrates these four measures and another potential element of DASD I/O time, titled "Other":



- C **IOSQ time.** IOSQ time is the time from the issuance of a STARTIO macro until the StartSubChannel (SSCH) instruction is issued. After the STARTIO macro is issued, the software determines whether the device is busy with *this system*; that is, whether there is an available Unit Control Block (UCB) for the device. If the device is not busy with *this system* (a UCB is available), the SSCH instruction is issued. However, if the device is busy with *this system*, the I/O request is queued. Thus, IOSQ time always means that the device is unable to handle additional requests from *this system*. (The emphasis on "this system" is explained in the below discussion of PEND time.)

This discussion of IOSQ time does not always apply to Parallel Access Volumes (PAVs)¹. With PAV devices, MVS creates multiple UCBs for each device, depending on how many "alias devices" have been defined. The multiple UCBs allow multiple active concurrent I/Os on a given device when the I/O requests originate from the same system². Using PAVs can dramatically improve I/O performance by nearly eliminating IOSQ.

¹PAV devices are available with Enterprise System Storage (ESS). With PAV devices, a "base device" address is defined, and a UCB is associated with this base address. "Alias device" addresses can be defined and UCBs are associated with the alias device addresses.

²Multiple Allegiance allows multiple active concurrent I/O operations on a given device when the I/O requests originate from different systems.

Please see Rule DAS250 for a more complete discussion of IOSQ time.

- C **PEND time.** PEND time is the time from the issuance of the StartSubChannel (SSCH) instruction until the device is selected by the control unit and physical positioning commands (such as seek and set sector) are transferred to the device. With modern fixed block architecture (FBA) devices, the PEND time ends when the physical positioning commands are presented to the *logical volume control block* within the control unit. The PEND time is caused by queuing for the path (wait for channel, wait for director port, wait for control unit, wait for device, or wait for “other” reasons)³.

The PEND time can be caused by the device being busy from *another system*. In this case, the system issuing the STARTIO macro (*this system*) would have no knowledge that the device was busy with another system. Rather, if a UCB were available for the device, the SSCH would be issued. However, the device could not necessarily be selected (unless multiple allegiance were available), since the device would be busy from another system.

Additionally, PEND time could accumulate even with PAV devices if the access were to an extent that was busy with another I/O operation from *this system*.

Please see Rule DAS230 for a more complete discussion of PEND time.

- C **DISC time.** DISC means that there is some delay that is often (but not always) associated with a mechanical movement during which the device disconnects from the control unit.

With legacy systems (e.g., 3380 drives attached to 3990-2 control units), the DISC time of most concern was associated with seek (arm movement) and rotational position sensing (time waiting for the disk platter to rotate to the location where desired data resides). Considerable performance improvement efforts were directed at reducing the seek activity and reducing the rotational position sensing

³PEND time is significantly reduced with FICON channels. FICON channels can have multiple I/O operations concurrently active, which reduces the potential PEND time caused by channel busy. There is no port busy time with FICON switches, and control unit time is significantly reduced. This statement regarding PEND time is not necessarily correct if a large number (more than 5) I/O operations are concurrently executing on a FICON channel. Dr. H. Pat Artis and Mr. Robert Ross have presented the results of research indicating that performance degrades significantly when more than 5 I/O operations (Open Exchanges) are concurrently active on a FICON channel (see “Understanding FICON Channel Path Metrics” at www.perfassoc.com).

(RPS)⁴ delays for the legacy systems. These two mechanical delays still exist for most modern *redundant array of independent disks* (RAID)⁵ systems, but their impact can not be directly reduced with normal methods.

With modern disks, data is cached into Actuator Level Buffers (ALBs), that contain data read from a track on the disk platter. Using ALBs can eliminate the RPS delays for records read on a particular track, since required data is read into the device buffer during a single rotation and stored until a path is available to transfer the data. However, if a record is to be read from a new track, some RPS delay could exist since the record would not be in the ALB, and must be read from the new track. Some initial RPS delay would apply in this case. This initial RPS delay is neither measured nor preventable.

Additionally, data is cached into increasingly large cache on the controller. For a read operation, desired data often is found in the cache. Write operations normally end as the data to be written is placed in non-volatile storage (NVS); and the storage processor writes the data to the device asynchronous with other activity (as a “back end” staging operation).

Consequently, DISC time for modern systems is a result of *cache read miss* operations, potentially back-end staging delay for write operations, peer-to-peer remote copy (PPRC) operations, and other miscellaneous reasons⁶. DISC time often can be very small with adequate cache. For example, there would be zero disconnect time for a cache read hit (the record was found in the cache).

Please see Rule DAS260 for a more complete discussion of DISC time.

- C **CONN time.** CONN time includes the data transfer time, but also includes protocol exchange⁷ (or “hand shaking”) between the various components at several stages of the I/O operation.

⁴RPS delays are caused by a path not being available when the required data came under a device read head. Since a path was not available, the data could not be read and another rotation of the platter was experienced until the data again came under the device read head. Multiple rotations might be required, depending on the busy level of the path.

⁵An array is an ordered collection of physical devices (disk drive modules) that are used to define logical volumes or devices.

⁶Artis has described a “sibling PEND” condition that results from collisions within the physical disk subsystem of RAID devices. See “Sibling PEND: Like a Wheel within a Wheel,” www.cmg.org/cmgpap/int449.pdf.

⁷Note that the protocol exchange occurs at multiple points in the normal I/O operation, even though it is shown only once in this exhibit.

For devices attached to paths that include parallel channels and ECON channels, the data transfer time is simply the number of bytes transferred divided by the transfer speed. This is because a parallel channel or ESCON channel can have only one data transfer operation in execution at one time.

For devices attached to paths that include FICON channels, the algorithm is more complicated. This primarily is because a FICON channel can perform multiple data transfer (read and write) operations at one time. The data packets for multiple read or write operations are interleaved (or multiplexed) in the FICON link. CONN time for an individual I/O begins with the first frame of data transferred and ends last frame of data transfer, even though data for other I/O operations might be transferred concurrently on the link. Consequently, if multiple data packets (representing data for multiple read or write operations) are interleaved on the FICON link, the elapsed time for any particular I/O operation can be elongated⁸ when compared with the elapsed time of the same I/O operation on an ESCON channel.

Please see Rule DAS240 for a more complete discussion of CONN time.

- C **OTHER time.** There are at least two other potential I/O delays for DASD: (1) waiting for the I/O completion interrupt to be serviced by a processor and (2) waiting for the I/O interrupt to be serviced by a domain under PR/SM. Neither potential I/O delay is expected to be of the magnitude of the four "standard" I/O delays. However, they can be significant in special circumstances.
- C Multi-processor configurations can use any processor to service an I/O interrupt. However, when a processor services an I/O interrupt, the processor's high-speed cache storage is no longer valid when control is returned to the interrupted task. Consequently, many of the processor's high-performance design features may be nullified.

A hardware feature allows processors to be disabled for I/O interrupts. With this method, only a small number (perhaps only one) processor is enabled for interrupt processing. Only this processor will have its high-speed cache storage disturbed by the task-switching required for interrupt processing, and only this

⁸The relative speed of a FICON channel is much higher than that of an ESCON channel. Consequently, the elapsed time of any particular I/O operation should be less on a FICON channel than on an ESCON channel, even if there are multiple I/O operations interleaving data. This statement regarding elapsed time is not necessarily correct if a large number (more than 5) I/O operations are concurrently executing on a FICON channel. Dr. H. Pat Artis and Mr. Robert Ross have presented the results of research indicating that performance degrades significantly when more than 5 I/O operations (Open Exchanges) are concurrently active on a FICON channel (see "Understanding FICON Channel Path Metrics" at www.perfassoc.com).

processor will periodically have its high-performance design features nullified. The disadvantage to this approach is that an interrupt may occur while the processor is busy servicing a previous interrupt.

If an interrupt is pending and no processor is enabled to service the interrupt, the interrupt must wait until a processor is available. This time should be insignificant, unless the system is processing a significantly large number of I/O operations. If the system is processing a large number of I/O operations (or if the I/O is particularly time-sensitive), the interrupt pending delay could pose performance problems.

After the processor completes processing for an I/O interrupt, it issues a Test Pending Interrupt (TPI) instruction to determine whether there are any interrupts pending. If an I/O interrupt is pending, the processor proceeds to service that interrupt.

The CPENABLE keyword in the IEAOPTxx member of SYS1.PARMLIB is used to specify the percent of I/O interrupts detected by the TPI instruction, compared with all I/O interrupts. When the percent exceeds the high threshold of the CPENABLE keyword, MVS enables another processor to handle pending I/O interrupts. If the percent falls below the low threshold of the CPENABLE keyword, MVS will disable a processor (to the point that only one processor is enabled). IBM's recommended setting for the CPENABLE keyword differs, depending on the level of processor.

- C MVS environments running under as a guest under VM or in a logical partition (LPAR) under PR/SM are subject to I/O interrupt delays. These delays can occur if another guest (for VM) or another domain is in its dispatch interval when the I/O interrupt completion is posted. The I/O interrupt remains pending until the guest or domain is dispatched. These delays have been estimated to be far more significant than might otherwise be expected.

Suggestion: There are no suggestions directly associated with this rule. Subsequent rules will analyze the device problems and attempt to determine the cause of poor performance.